REMARKS

Claims 48-54 are added herein. Claims 1-54 are pending. Claims 1-3, 6, 14, 26-28 and 37 are amended herein. No new matter is added as a result of the claim amendments.

Objection

The specification of the present application is objected to because it does not include reference to the parent application. The specification is amended herein to overcome this objection.

Double Patenting Rejection

Claim 1 is provisionally rejected under the judicially created (nonstatutory) doctrine of obviousness-type double patenting as being unpatentable over Claim 1 of US Patent No. 6,421,059. A terminal disclaimer in compliance with 37 CFR § 1.321 is being submitted concurrent with the instant response, thereby obviating the double patenting rejection.

Specification

The specification is objected to because of the informalities cited in the instant Office Action. The specification is amended herein to correct those informalities.

Claim Objections

Claim 6 is objected to because of the informality cited in the instant Office Action. Claim 6 is amended herein to correct that informality.

112 Rejection

Claims 21-23 and 44-45 are rejected under 35 U.S.C. § 112, first paragraph. Specifically, the Examiner states that "The glyph information register appears in the claims, but it is not described in the specification."

The Examiner is respectfully directed to, for example, page 11, lines 4 and 5, of the instant specification, which states "The graphics controller 400 also includes a register 410 that provides the glyph information of a particular character to an expansion block 402 of the graphics controller 400." That is, register 410 that provides the glyph information is one embodiment of the glyph information register of the claims. As such, Applicant respectfully submits that the rejection of Claims 21-23 and 44-45 under 35 U.S.C. § 112, first paragraph, is traversed.

102 Rejection

Claims 1-47 are rejected under 35 U.S.C. § 102(b) as being anticipated by Lobodzinski et al. ("Lobodzinski;" US 5,734,873). The Applicant has reviewed the cited reference and respectfully submits that the present invention as recited in Claims 1-47 is not anticipated or shown by Lobodzinski.

Applicant respectfully submits that Lobodzinski does not show or suggest "a graphics controller coupled to the memory, the graphics controller accessing a font array of the data structure, the graphics controller comprising memory for holding information read from the font array" as recited in independent Claim 1 (emphasis added). In addition, Applicant respectfully submits that Lobodzinski does not show or suggest "placing the information read from the font array in

memory resident on a graphics controller" as recited in independent Claim 26 (emphasis added). The Examiner is directed to Figure 2 of Lobodzinski and the accompanying discussion. Applicant respectfully notes that there is no showing or suggestion in Lobodzinski that graphics engine 48 places information read from character font information 62 into memory resident on the graphics engine 48.

Therefore, Applicant respectfully submits that Lobodzinski does not show or suggest the present invention as recited in independent Claims 1 and 26. Accordingly, Applicant respectfully submits that these claims traverse the Examiner's basis for rejection under 35 U.S.C. § 102(b) and are in condition for allowance. Claims 2-25 are dependent on Claim 1, and Claims 27-47 are dependent on Claim 26. As such, Applicant respectfully submits that Claims 2-25 and 27-47 also traverse the Examiner's basis for rejection under 35 U.S.C. § 102(b) as these claims are dependent on allowable base claims and recite additional limitations.

CONCLUSION

In light of the above remarks, Applicant respectfully requests reconsideration of the rejected Claims.

Based on the arguments presented above, Applicant respectfully asserts that Claims 1-54 overcome the rejections of record and, therefore, Applicant respectfully solicits allowance of these Claims.

Applicant has reviewed the references that were cited but not relied upon. Applicant did not find these references to show or suggest the present claimed invention: US 6,236,390 and US 5,870,084.

The Examiner is invited to contact Applicant's undersigned representative if the Examiner believes such action would expedite resolution of the present Application.

Respectfully submitted,

WAGNER, MURABITO & HAO LLP

Date: 2/6/03

William A. Zarbis
Reg. No. 46,120

Two North Market Street
Third Floor
San Jose, California 95113
(408) 938-9060

VERSION WITH MARKINGS TO SHOW CHANGES MADE

IN THE SPECIFICATION

Please amend the specification as follows.

Please insert the following paragraph on page 1 at line 1:

--RELATED U.S. APPLICATION

This application is a Continuation of the copending patent application, US Patent No. 6,421,059, with issue date July 16, 2002, assigned to the assignee of the present application and hereby incorporated by reference. --

Please amend the paragraphs starting on page 6, line 19, and ending on page 7, line 19, as follows:

-- Figure 3 illustrates a glyph or a compressed font 300 produced in accordance with the second conventional technique. Pixels are represented by a bit of information (i.e., a 1 or a[n] 0). As is seen, each bit 302 represents one of the bytes of the information that was shown previously in Figure 2. Accordingly, rather than having to provide 35 bytes of information in the case of an 8 bit frame buffer, in this environment the CPU 12 (Figure 1) would only need to provide 35 bits across the bus 16, which translates into [a] 4 3/8 bytes of information. These 4 3/8 bytes of information therefore can be decompressed utilizing circuitry in the graphics controller 24.

Figure 4 illustrates a graphics controller 24 and a frame buffer 26 in accordance with the second conventional technique. Typically, the graphics controller 24 will include an expansion block 302 which will render the pixel into the frame buffer 26 based upon the glyphs. The expansion block 302 includes first and second registers 312 and 314. A multiplexer 310 within the expansion block 302 is utilized to select between the two registers 312 and 314 based upon the glyph. The selection of the registers 312 and 314 is based upon the bits provided by the CPU 12 (Figure 1). Accordingly, if the bit provided by the CPU 12 is a foreground color, the multiplexer 310 selects the register 312 which would expand the bit to a plurality of bits (i.e., 8 bits, 16 bits or 32 bits) and those bits are then provided into the frame buffer 26 as a foreground color. On the other hand, if the bit provided is background color, the multiplexer 310 selects register 314 which would expand the bit into a plurality of bits and those bits are then provided to the frame buffer 26 as a background color. As further improvement in the second embodiment, a so-called transparent mode can be provided. In this mode, a multiplexer 311 allows for the background color to be provided from the frame buffer 26 via select line 315 and input 313. For example, if the background color is 00 then the frame buffer automatically loads zeros for the background color of that particular character. In so doing, rendering time is further reduced. Accordingly the character 100 that is rendered in the frame buffer is exactly the same as that shown in Figure 1. --

Please amend the paragraph on page 9, starting at line 4, as follows:

-- To discuss the present invention in the context of a preferred embodiment, refer now to Figure 5 and the accompanying discussion. Figure 5 is a block diagram of a graphics controller 400 and a frame buffer 450 in accordance

with the present invention. The graphics controller includes an expansion unit 402 that has similar components to that of expansion unit 302 of Figure 4. The memory 450 includes a data structure 451. The data structure 451 includes a plurality of font arrays 460 and 462. It should be understood that there could be any number of font arrays in the data structure 451. As is seen, in font array 462 the [size] pitch of the font characters [are] is larger than the [size] pitch of the font characters in font array 460. Hence, for each font array 460 and 462 a different index and a different font pointer [is] are required to allow the graphics controller to access the associated font characters. Accordingly, a font pointer is changed when, for example, the font is changed. Each of the font arrays 460 and 462 include information concerning the size of each character and an index that indicates the location of each font character within the font array. The font characters can be either full sized fonts or glyphs. The information within the font array is utilized by the graphics controller 400 to allow the graphics controller 400 to render a particular character retrieved from the memory 450. --

Please amend line 19 on page 10 as follows:

-- SIZE Width Register 420 --

Please amend the paragraph on page 12, beginning at line 9, as follows:

-- In an example, all the user needs is to load the index value, the x value and the y value for the character that is to be rendered, where the index is an ASCII character number. In a preferred embodiment, the x value can be 12 bits, the y value can be 12 [bites] bits and the index value can be 8 bits. In so doing, one

32 bit transfer from the CPU to the graphics controller is all that is needed to render the character. --

IN THE CLAIMS

Please amend the claims as follows.

1. (Once Amended) A system for rendering fonts [into a memory], the system comprising:

a memory having stored therein a data structure, the [located within a memory or other memories; the] data structure including at least one font array; and

a graphics controller coupled to the memory, the graphics controller [for] accessing [said at least one] a font array of the data structure [and for rendering characters of said at least one font array], the graphics controller comprising memory for holding information read from the font array [into the appropriate locations of a memory or other memories].

2. (Once Amended) The system of claim 1 wherein [any of the memories] the memory comprises a frame buffer.

3. (Once Amended) The system of claim 1 wherein [any of the memories] the memory comprises a system memory.

6. (Once Amended) The system of claim 4 in which each of the characters comprises a plurality of bits per pixel[s].

14. (Once Amended) The system of claim [1] 7 in which the graphics controller comprises:

a set of registers for utilizing the information within the plurality of font arrays such that font characters can be efficiently retrieved [from memory] and [can then be] rendered [in the memory].

26. (Once Amended) A method for rendering fonts [into a memory], the method comprising [the steps of]:

[(a) providing] accessing a data structure located in a memory [or other memories; the], the data structure including at least one font array;

[(b) accessing said at least one] reading information from a font array of the data structure; and

placing the information read from the font array in memory resident on a graphics controller.

[(c) rendering characters of said at least one font array into the appropriate locations of a memory or other memories.]

27. (Once Amended) The method of claim 26 wherein [any of the memories] the memory comprises a frame buffer.

28. (Once Amended) The method of claim 26 wherein [any of the memories] the memory comprises a system memory.

37. (Once Amended) The method of claim [26] 32 in which the graphics controller includes:

a set of registers for utilizing the information within the plurality of font arrays such that font characters can be efficiently retrieved [from memory and can then be rendered in the memory] and rendered.

Please add the following new claims:

48. (New) A system for rendering characters, said system comprising:

a memory having stored therein a data structure, said data structure comprising glyph information for each of a plurality of characters, said data structure also comprising size width information and size height information for each of said characters; and

a graphics controller coupled to said memory;

wherein glyph information for a character to be rendered, said size width information and said size height information are read to registers of said graphics controller from said data structure, said graphics controller using said glyph information to render said character in a frame buffer according to said size width and size height information.

49. (New) The system of Claim 48 wherein said memory comprises a portion of said frame buffer.

50. (New) The system of Claim 48 wherein said memory comprises a plurality of data structures, each of said data structures corresponding to a particular character font.

51. (New) The system of Claim 48 wherein each of said characters in said data structure is identified by an index.

52. (New) The system of Claim 51 wherein said graphics controller receives a value for said index.

53. (New) The system of Claim 48 wherein said graphics controller receives a value that points to said data structure.

54. (New) The system of Claim 48 wherein said graphics controller receives values for the horizontal and vertical locations in said frame buffer for rendering said character.